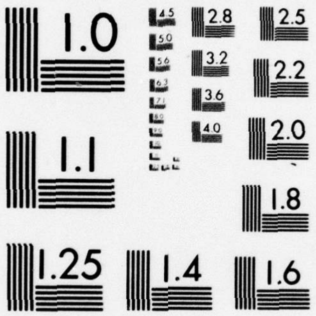
AD-A069 876 COMPUTER SCIENCES CORP SILVER SPRING MD SYSTEM SCIEN--ETC F/G 9/2
CENTRAL FLOW CONTROL MANAGEMENT AIDS (MA) COMPONENT PDL/PROLOGU--ETC(U)
JAN 79 DOT-FA77WA-3955
UNCLASSIFIED CSC/SD-78/6160 FAA-RD-79-37 NL

| OF |
AD
A069876

END
DATE
FILMED
7 -79
DDC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

Report No. FAA-RD-79-37

CENTRAL FLOW CONTROL MANAGEMENT AIDS CODE AUDITOR (MACA) PDL AND PROLOGUE

January 1979

Final Report

Document is available to the U.S. public through
the National Technical Information Service,
Springfield, Virginia 22161.

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
FEDERAL AVIATION ADMINISTRATION
Systems Research & Development Service
Washington, D.C. 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Technical Report Documentation Page

19

1. Report No. 18 FAA-RD-79-37	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle 6 Central Flow Control Management Aids (MA) Component PDL/Prologue.Analysis User's Manual.		5. Report Date 11 Jan[redacted]79
		6. Performing Organization Code
		8. Performing Organization Report No. 14 CSC/SD-78/6160
7. Author(s) Computer Sciences Corporation		
9. Performing Organization Name and Address Computer Sciences Corporation System Sciences Division 8728 Colesville Road Silver Spring, Maryland 20910		10. Work Unit No. (TRAIS)
		11. Contract or Grant No. 15 DOT-FA77WA-3955
		13. Type of Report and Period Covered 9 Final Rep[redacted]t.
12. Sponsoring Agency Name and Address U.S. Department of Transportation Federal Aviation Administration Systems Research and Development Service Washington, D.C. 20591		14. Sponsoring Agency Code ARD-102

15. Supplementary Notes

12 27p.

16. Abstract

This document describes the functions of the PDL/Prologue Analysis programs and details the procedures required to exercise them.

These programs provide analysis of module prologues and Program Design Language (PDL) as part of the Central Flow Control (CFC) Quality Assurance (QA) program, which assures accuracy, completeness, and adherence to project standards and guidelines.

(Title on t.page is correct per phone convesation/w M. Swan DOT)

17. Key Words CENTRAL FLOW CONTROL CODE ANALYZER TEST SPECIFICATION GENERATOR	18. Distribution Statement This document is available to the public through the National Technical Information Service (NTIS), Springfield, Virginia 22151

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	24	

Reproduction of completed page authorized

408 479

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	*2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

*1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures, Price $2.25, SD Catalog No. C13.10:286.

inches
1
2
3
4
5
6
7
8
9
cm
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

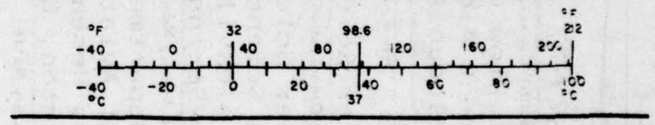


TABLE OF CONTENTS

Section 1 - Introduction 1-1

1.1 Purpose and Scope. 1-1
1.2 Background Information 1-1
1.3 References 1-1

Section 2 - Program Operation. 2-1

2.1 Program Operation Overview 2-1

Section 3 - Program Inputs 3-1

3.1 Control Cards. 3-1
3.1.1 Control Card Examples. 3-2
3.2 Data Cards 3-4
3.3 Data Sets. 3-4
3.3.1 Control Card Set 3-4
3.3.2 Prologue/PDL Library Set 3-4
3.3.3 Prologue/PDL Report Set. 3-4
3.3.4 Diagnostic Error Message Set 3-5
3.4 Job Control Language 3-5
3.4.1 MACA Procedure 3-5
3.4.2 How to Use MACA Procedure. 3-6

Section 4 - Program Outputs. 4-1

Section 5 - Diagnostics. 5-1

LIST OF ILLUSTRATIONS

FIGURE PAGE

1-1 Module Prologue. 1-2

1-2 Program Design Language. 1-3

1-3 Test Specification Chart 1-4

4-1 Control Card Listing 4-2

4-2 Prologue/PDL Report. 4-3

4-3 Test Specification Chart 4-4

SECTION 1 - INTRODUCTION

1.1 PURPOSE AND SCOPE

The Automated Code Auditor is a support program which provides these services:

1. Analysis of module prologues for completeness and adherence to standards
2. Parsing of Program Design Language (PDL) for proper expression of the design
3. Generation of Test Specification charts from PDL.

The user's manual provides information on the set-up and execution of the Code Auditor.

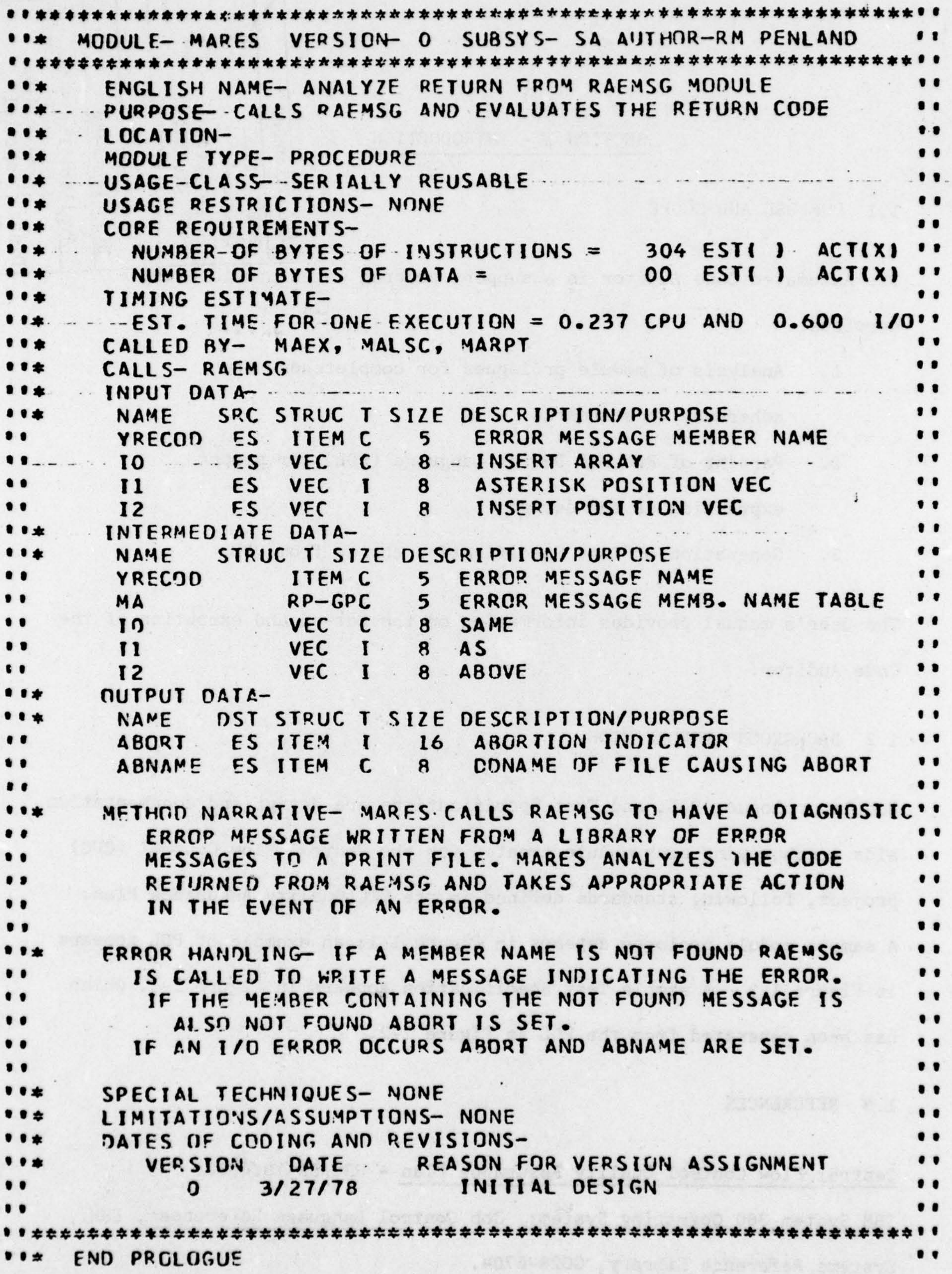
1.2 BACKGROUND INFORMATION

Module prologue, PDL, and Test Specifications are design and documentation aids accompanying each module created for the Central Flow Control (CFC) project, following standards defined in the CFC Quality Assurance Plan. A sample module prologue appears in Figure 1-1; an example of PDL appears in Figure 1-2. A sample Test Specification appears in Figure 1-3, which has been generated from the PDL in Figure 1-2.

1.3 REFERENCES

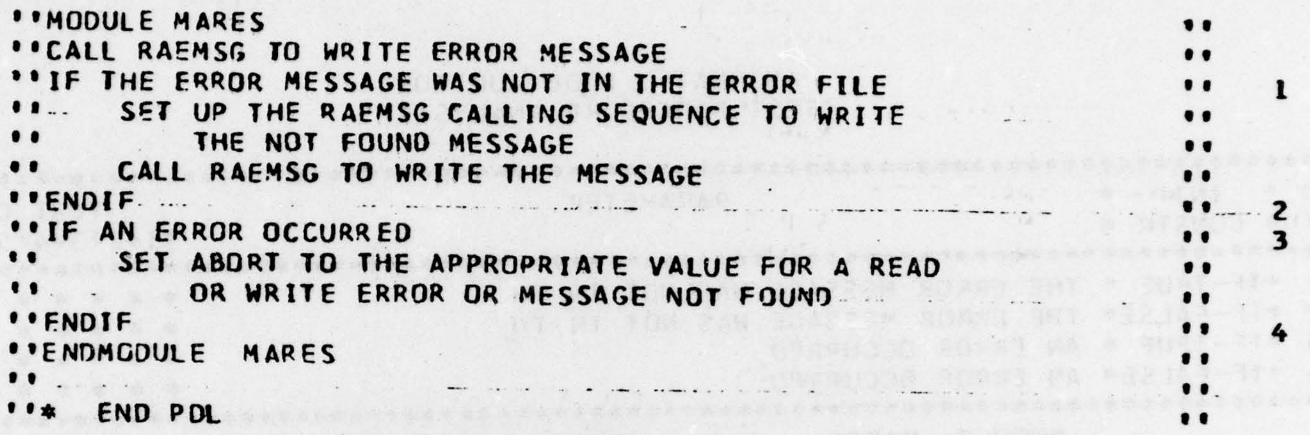
<u>Central Flow Control Quality Assurance Plan</u> - CSC/SD-78/6060.

<u>IBM System 360 Operating System: Job Control Language References</u>, IBM. Systems Reference Library, GC28-6704.



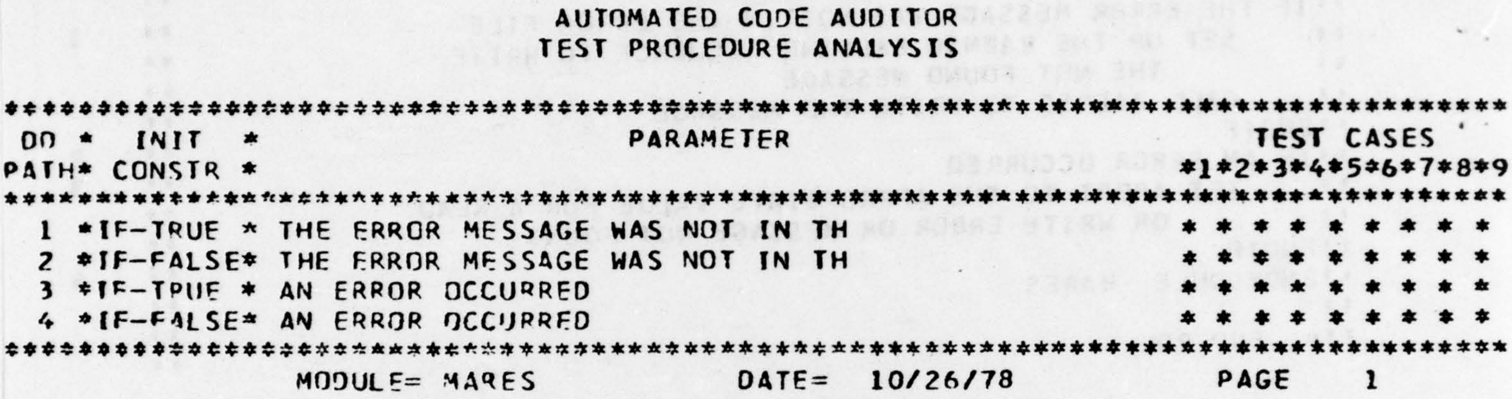
```
••*************************************************************************••
••*  MODULE- MARES  VERSION- 0  SUBSYS- SA AUTHOR-RM PENLAND               ••
••*************************************************************************••
••*     ENGLISH NAME- ANALYZE RETURN FROM RAEMSG MODULE                    ••
••*     PURPOSE- CALLS RAEMSG AND EVALUATES THE RETURN CODE                ••
••*     LOCATION-                                                          ••
••*     MODULE TYPE- PROCEDURE                                             ••
••*     USAGE-CLASS- SERIALLY REUSABLE                                     ••
••*     USAGE RESTRICTIONS- NONE                                           ••
••*     CORE REQUIREMENTS-                                                 ••
••*       NUMBER-OF BYTES OF INSTRUCTIONS =    304 EST( )  ACT(X)          ••
••*       NUMBER OF BYTES OF DATA =            00  EST( )  ACT(X)          ••
••*     TIMING ESTIMATE-                                                   ••
••*       EST. TIME FOR ONE EXECUTION = 0.237 CPU AND  0.600  I/O          ••
••*     CALLED BY-  MAEX, MALSC, MARPT                                     ••
••*     CALLS- RAEMSG                                                      ••
••*     INPUT DATA-                                                        ••
••*      NAME    SRC STRUC T SIZE DESCRIPTION/PURPOSE                      ••
••       YRECOD  ES  ITEM C   5   ERROR MESSAGE MEMBER NAME                ••
••       I0      ES  VEC  C   8   INSERT ARRAY                             ••
••       I1      ES  VEC  I   8   ASTERISK POSITION VEC                    ••
••       I2      ES  VEC  I   8   INSERT POSITION VEC                      ••
••*     INTERMEDIATE DATA-                                                 ••
••*      NAME    STRUC T SIZE DESCRIPTION/PURPOSE                          ••
••       YRECOD      ITEM C   5  ERROR MESSAGE NAME                        ••
••       MA          RP-GPC   5  ERROR MESSAGE MEMB. NAME TABLE            ••
••       I0          VEC  C   8  SAME                                      ••
••       I1          VEC  I   8  AS                                        ••
••       I2          VEC  I   8  ABOVE                                     ••
••*     OUTPUT DATA-                                                       ••
••*      NAME    DST STRUC T SIZE DESCRIPTION/PURPOSE                      ••
••       ABORT   ES ITEM  I   16  ABORTION INDICATOR                       ••
••       ABNAME  ES ITEM  C   8   DDNAME OF FILE CAUSING ABORT             ••
••                                                                         ••
••*     METHOD NARRATIVE- MARES CALLS RAEMSG TO HAVE A DIAGNOSTIC          ••
••         ERROR MESSAGE WRITTEN FROM A LIBRARY OF ERROR                   ••
••         MESSAGES TO A PRINT FILE. MARES ANALYZES THE CODE               ••
••         RETURNED FROM RAEMSG AND TAKES APPROPRIATE ACTION               ••
••         IN THE EVENT OF AN ERROR.                                       ••
••                                                                         ••
••*     ERROR HANDLING- IF A MEMBER NAME IS NOT FOUND RAEMSG               ••
••        IS CALLED TO WRITE A MESSAGE INDICATING THE ERROR.               ••
••         IF THE MEMBER CONTAINING THE NOT FOUND MESSAGE IS               ••
••           ALSO NOT FOUND ABORT IS SET.                                  ••
••        IF AN I/O ERROR OCCURS ABORT AND ABNAME ARE SET.                 ••
••                                                                         ••
••*     SPECIAL TECHNIQUES- NONE                                           ••
••*     LIMITATIONS/ASSUMPTIONS- NONE                                      ••
••*     DATES OF CODING AND REVISIONS-                                     ••
••*       VERSION   DATE      REASON FOR VERSION ASSIGNMENT                ••
••           0     3/27/78       INITIAL DESIGN                            ••
••                                                                         ••
••*************************************************************************••
••*  END PROLOGUE                                                          ••
```

Figure 1-1. Module Prologue



```
''MODULE MARES                                                   ''
''CALL RAEMSG TO WRITE ERROR MESSAGE                             ''
''IF THE ERROR MESSAGE WAS NOT IN THE ERROR FILE                 ''    1
''    SET UP THE RAEMSG CALLING SEQUENCE TO WRITE                ''
''          THE NOT FOUND MESSAGE                                ''
''    CALL RAEMSG TO WRITE THE MESSAGE                           ''
''ENDIF                                                          ''    2
''IF AN ERROR OCCURRED                                           ''    3
''    SET ABORT TO THE APPROPRIATE VALUE FOR A READ              ''
''          OR WRITE ERROR OR MESSAGE NOT FOUND                  ''
''ENDIF                                                          ''    4
''ENDMODULE  MARES                                               ''
''                                                               ''
''*  END PDL                                                     ''
```

Figure 1-2. Program Design Language



```
                         AUTOMATED CODE AUDITOR
                        TEST PROCEDURE ANALYSIS
********************************************************************************
 DO *  INIT  *                  PARAMETER                         TEST CASES
PATH* CONSTR *                                                 *1*2*3*4*5*6*7*8*9
********************************************************************************
  1 *IF-TRUE * THE ERROR MESSAGE WAS NOT IN TH                  * * * * * * * * *
  2 *IF-FALSE* THE ERROR MESSAGE WAS NOT IN TH                  * * * * * * * * *
  3 *IF-TRUE * AN ERROR OCCURRED                                * * * * * * * * *
  4 *IF-FALSE* AN ERROR OCCURRED                                * * * * * * * * *
********************************************************************************
           MODULE= MARES               DATE=  10/26/78              PAGE    1
```

Figure 1-3. Test Specification Chart

SECTION 2 - PROGRAM OPERATION

2.1 PROGRAM OPERATION OVERVIEW

Analysis of module prologues is accomplished through simple parsing of keywords and associated text. Prologue keywords fall into two classes: major and minor. Referring to the fourth line of Figure 1-1, ENGLISH is a major keyword and NAME- immediately thereafter is a minor keyword. Once a major keyword is recognized by the Code Auditor, comparison is made for the associated minor keyword(s) and descriptive text. Keywords must begin in specific card columns and must be spelled correctly. Any character string not meeting these criteria is considered descriptive text. Errors in card column alignment or missing keywords are flagged with an error message.

The Program Design Language (PDL) logical constructs are described in Section 3.2.1.2 of the CFC Quality Assurance Plan. Parsing of PDL by the Code Auditor involves the following groups of keywords:

```
IF/ENDIF
IF/ELSE/ENDIF
DO-WHILE/ENDDO
REPEAT/UNTIL
DO-CASE/ENDDO.
```

These groups of keywords are essentially delimeters of logical flow. For example, whenever the keyword DO-CASE is recognized, there must also appear a subsequent ENDDO beginning in the same card column. This relationship applies to all the keyword groups. Figure 1-2 contains an example of PDL.

The Test Specification chart is generated during the parsing of PDL. (Refer to Figure 1-3, which is a Test Specification Chart generated from the PDL in Figure 1-2.) Decision-to-decision paths are created as PDL keywords are recognized. The Code Auditor assigns a number to each of these, which is printed both on the left side of the Test Specification chart and on the right side of the PDL report. The first thirty characters following an IF, DO-WHILE, and DO-CASE are printed on the Test Specificaation Chart to help identify the decision-to-decision paths.

SECTION 3 - PROGRAM INPUTS

3.1 CONTROL CARDS

The control card data set is read from the SYSIN file and should appear as follows:

```
START  $
  .
  .
  .
MODULE CARDS .
  .
  .
  .
TERM   $
```

The information on control cards may appear between columns 1 and 72 inclusive. The $ delimeter is optional on each card; if present, the remainder of the card is ignored and may be used for comments.

The START card signals the beginning of the control card set and should not appear elsewhere. The format is:

```
START    $
```

The MODULE card identifies the name and specifies options for a module to be analyzed. At least one and not more than one hundred of these cards must appear in the control card set. The format:

```
MODULE = name, option, option, option  $
```

VALUE FIELD	LENGTH (BYTES)	VALUES	REQUIRED/ OPTIONAL	DESCRIPTION
name	8		R	name of a member in the Prologue/PDL library
option	4	LPRL	O	list prologue only
		LSRC	O	list PDL only
		LTPA	O	generate and list Test Specification charts

Special attention must be given to the LTPA parameter. If LTPA is specified on the first MODULE card, Test Specification charts will be generated and listed for all modules. Similarly, if LTPA is not present on the first MODULE card, no Test Specification charts will be listed for any module.

If no optional parameters accompany a module name, only diagnostics will be listed. If LPRL and LSRC are present, the listing includes both Prologue and PDL.

The TERM card identifies the end of the control card set and should not appear elsewhere. The format is:

```
TERM    $
```

3.1.1 Control Card Examples

To generate Test Specification Charts

```
START    $
  MODULE = MOD1, LTPA  $
  MODULE = MOD2    $
TERM     $
```

Test Specification Charts are to be generated and listed for modules MOD1 and MOD2, from their respective PDL. Their prologues and PDL will be analyzed and only diagnostics (if any) will be listed.

To Analyze and List the Prologue:

```
START    $
  MODULE = MOD1, LPRL  $
TERM     $
```

The MOD1 Prologue is to be analyzed and listed with diagnostics.

To Analyze and List the PDL:

```
START    $
  MODULE = MOD2, LSRC  $
TERM     $
```

The MOD2 PDL is to be listed. The MOD2 Prologue will be analyzed and diagnostics (if any) will be listed.

For All Code Auditor Services:

```
START    $
  MODULE = MOD1, LPRL, LSRC, LTPA  $
  MODULE = MOD2, LSRC, LPRL  $
  MODULE = MOD3, LPRL, LSRC  $
TERM     $
```

The prologues and PDL of modules MOD1, MOD2, and MOD3 are to be analyzed and listed with diagnostics. Test Specification Charts are to be generated from PDL and listed for all three modules.

For Minimal Listing:

```
START    $
  MODULE = MOD1  $
TERM     $
```

The PDL and prologue of MOD1 is to be analyzed, but not listed. Only diagnostics (if any) are to be listed.

3.2 DATA CARDS

Data cards are not used by the Code Auditor.

3.3 DATA SETS

The Automated Code Auditor uses the following data sets:

- Control Card Set
- Prologue/PDL Library Set
- Prologue/PDL Report Set
- Diagnostic Error Message Set

3.3.1 Control Card Set

Control cards are input through the sequential file defined by the SYSIN DD card. Logical records are 80 bytes long and can be blocked by the user. No specific hardware device must be used for the control card file.

3.3.2 Prologue/PDL Library Set

This input file is defined by the DATAIN DD card. The prologue/PDL library must be a partitioned data set whose members are modules to be analyzed. Each module should be composed of a prologue followed by PDL. The logical record length is 80 bytes, and the records should be blocked.

3.3.3 Prologue/PDL Report Set

The Automated Code Auditor has only one output, the Prologue/PDL Report Set, identified by the MARPTLST DD card. Logical record length is 133 bytes. As this file is intended to be a print file, the record format is FBA, fixed-length blocked records with ANSI printer control characters.

3.3.4 Diagnostic Error Message Set

This file is a partitioned data set whose members each contain text for a diagnostic error message. This input file, identified via the MAEMSG DD card, is crucial to the Code Auditor and must be disk-resident whenever that program is executed. Each member is one 80 byte fixed-length record.

3.4 JOB CONTROL LANGUAGE

The following sections describe the Job Control Language (JCL) available for execution of MACA and its use.

3.4.1 MACA Procedure

The MACA catalogued procedure is a member of SYS2.PROCLIB, and contains the following:

```
//MACA PROC DATAIN=
//   EXEC PGM=MACA,REGION=170K
//STEPLIB  DD DSN=MA.LIB.LOAD,DISP=SHR
//DATAIN DD DSN=&DATAIN,DISP=SHR
//MARPTLST   DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=133,BLKSIZE=3458)
//MAEMSG DD DSN=SPCX.LIB.ERROR.CURRENT,DISP=SHR
//MASUMLST DD DUMMY,DCB=*.MARPTLST
```

where the symbolic parameter DATAIN specifies the user's prologue/PDL library described in Section 3.3.2.

The DDNAMES are used as follows:

DDNAME	USAGE
STEPLIB	Library where MACA load module resides.
DATAIN	User's Prologue/PDL Library. This file must be catalogued.
MARPTLST	Prologue/PDL report file .
MAEMSG	Diagnostic Error Message Library. This file must be catalogued.
MASUMLST	Although this report file is not created, the DD card must be present as shown.

3.4.2 How to Use MACA Procedure

The example below illustrates the job control statements required to use the Code Auditor:

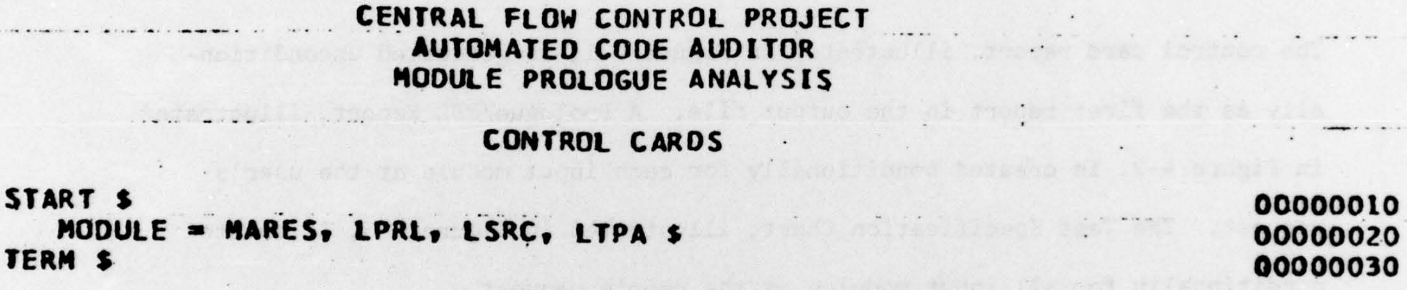
```
//  EXEC  MACA,DATAIN='user.catalogued.pds'
//SYSIN  DD  *
          .
          .
          .
     CONTROL CARDS
          .
          .
          .
/*
```

SECTION 4 - PROGRAM OUTPUTS

The Automated Code Auditor creates one output, a print file, which contains the following:

- Control Card Listing
- Module Prologue/PDL Listing
- Test Specification Chart.

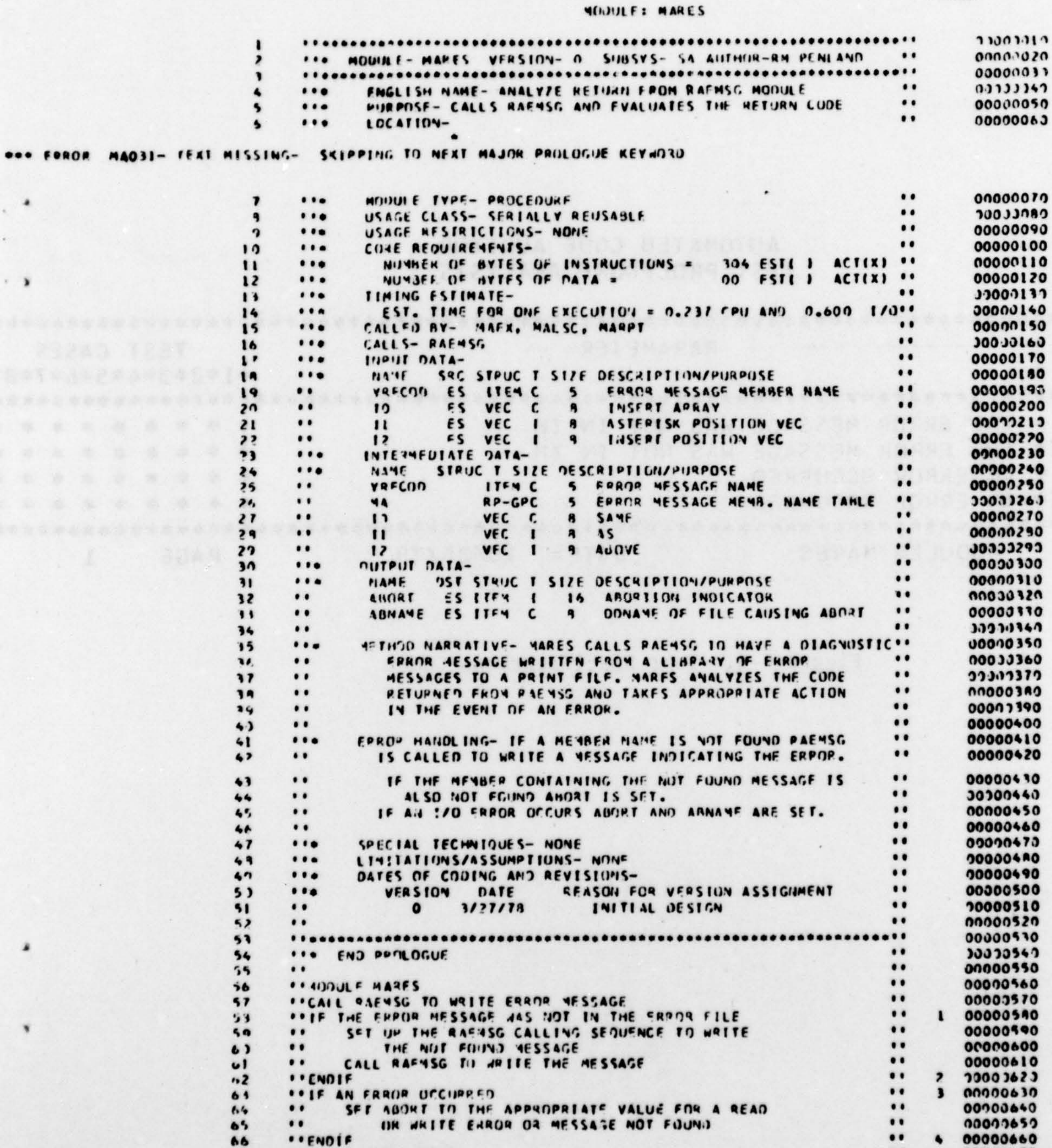
The control card report, illustrated in Figure 4-1, is generated unconditionally as the first report in the output file. A Prologue/PDL Report, illustrated in Figure 4-2, is created conditionally for each input module at the user's request. The Test Specification Chart, illustrated in Figure 4-3, is created conditionally for all input modules at the user's request.



```
                      CENTRAL FLOW CONTROL PROJECT
                         AUTOMATED CODE AUDITOR
                       MODULE PROLOGUE ANALYSIS

                              CONTROL CARDS
START $                                                         00000010
   MODULE = MARES, LPRL, LSRC, LTPA $                           00000020
TERM $                                                          00000030
```

Figure 4-1. Control Card Listing



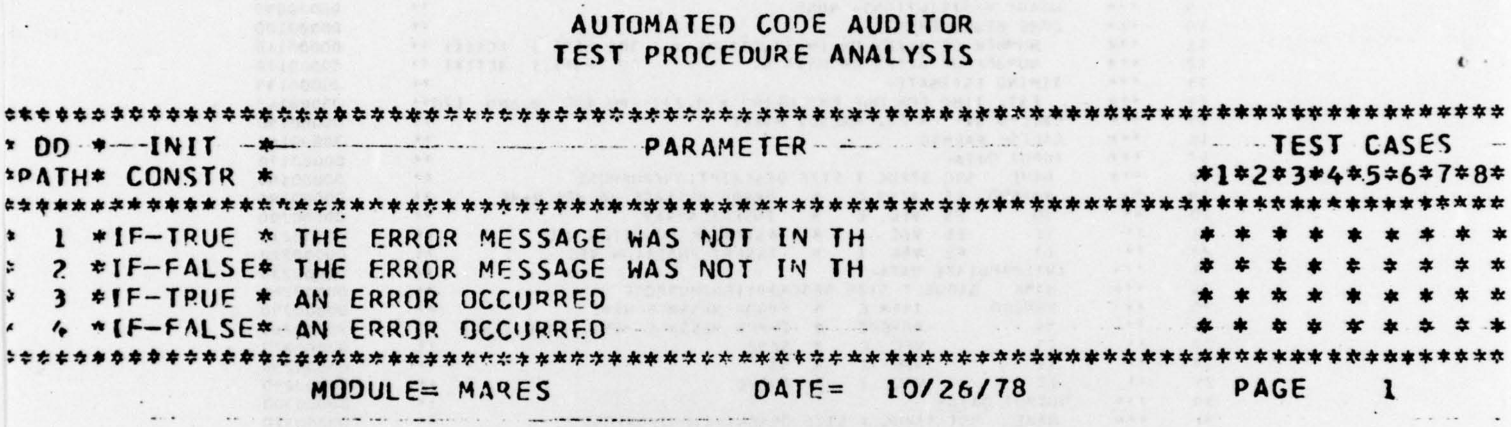
```
                                    MODULE: MARES

          1   *********************************************************************          00000010
          2   ***  MODULE- MARES  VERSION- 0  SUBSYS- SA AUTHOR-RM PENLAND        **          00000020
          3   *********************************************************************          00000030
          4   ***     ENGLISH NAME- ANALYZE RETURN FROM RAEMSG MODULE               **          00000040
          5   ***     PURPOSE- CALLS RAEMSG AND EVALUATES THE RETURN CODE           **          00000050
          6   ***     LOCATION-                                                     **          00000060
                      *
*** ERROR  MA031- TEXT MISSING-  SKIPPING TO NEXT MAJOR PROLOGUE KEYWORD

          7   ***     MODULE TYPE- PROCEDURE                                        **          00000070
          8   ***     USAGE CLASS- SERIALLY REUSABLE                                **          00000080
          9   ***     USAGE RESTRICTIONS- NONE                                      **          00000090
         10   ***     CORE REQUIREMENTS-                                            **          00000100
         11   ***       NUMBER OF BYTES OF INSTRUCTIONS =    304 EST( )  ACT(X)     **          00000110
         12   ***       NUMBER OF BYTES OF DATA =             00  EST( )  ACT(X)    **          00000120
         13   ***     TIMING ESTIMATE-                                              **          00000130
         14   ***       EST. TIME FOR ONE EXECUTION = 0.237 CPU AND  0.600  I/O     **          00000140
         15   ***     CALLED BY-  MAEX, MALSC, MARPT                                **          00000150
         16   ***     CALLS- RAEMSG                                                 **          00000160
         17   ***     INPUT DATA-                                                   **          00000170
         18   ***      NAME    SRC STRUC T SIZE DESCRIPTION/PURPOSE                 **          00000180
         19   **       VRECOD  ES  ITEM C   5   ERROR MESSAGE MEMBER NAME           **          00000190
         20   **       I0       ES  VEC  C   8   INSERT ARRAY                       **          00000200
         21   **       I1       ES  VEC  I   8   ASTERISK POSITION VEC              **          00000210
         22   **       I2       ES  VEC  I   9   INSERT POSITION VEC                **          00000220
         23   ***     INTERMEDIATE DATA-                                            **          00000230
         24   ***      NAME    STRUC T SIZE DESCRIPTION/PURPOSE                     **          00000240
         25   **       VRECOD        ITEM C   5  ERROR MESSAGE NAME                 **          00000250
         26   **       MA            RP-GPC   5  ERROR MESSAGE MEMB. NAME TABLE     **          00000260
         27   **       I0            VEC  C   8  SAME                               **          00000270
         28   **       I1            VEC  I   8  AS                                 **          00000280
         29   **       I2            VEC  I   9  ABOVE                              **          00000290
         30   ***     OUTPUT DATA-                                                  **          00000300
         31   ***      NAME    DST STRUC T SIZE DESCRIPTION/PURPOSE                 **          00000310
         32   **       ABORT    ES ITEM  I   16  ABORTION INDICATOR                 **          00000320
         33   **       ABNAME   ES ITEM  C   8   DDNAME OF FILE CAUSING ABORT       **          00000330
         34   **                                                                    **          00000340
         35   ***     METHOD NARRATIVE- MARES CALLS RAEMSG TO HAVE A DIAGNOSTIC     **          00000350
         36   **         ERROR MESSAGE WRITTEN FROM A LIBRARY OF ERROR              **          00000360
         37   **         MESSAGES TO A PRINT FILE. MARES ANALYZES THE CODE          **          00000370
         38   **         RETURNED FROM RAEMSG AND TAKES APPROPRIATE ACTION          **          00000380
         39   **         IN THE EVENT OF AN ERROR.                                  **          00000390
         40   **                                                                    **          00000400
         41   ***     ERROR HANDLING- IF A MEMBER NAME IS NOT FOUND RAEMSG          **          00000410
         42   **        IS CALLED TO WRITE A MESSAGE INDICATING THE ERROR.          **          00000420

         43   **         IF THE MEMBER CONTAINING THE NOT FOUND MESSAGE IS          **          00000430
         44   **           ALSO NOT FOUND ABORT IS SET.                             **          00000440
         45   **        IF AN I/O ERROR OCCURS ABORT AND ABNAME ARE SET.            **          00000450
         46   **                                                                    **          00000460
         47   ***     SPECIAL TECHNIQUES- NONE                                      **          00000470
         48   ***     LIMITATIONS/ASSUMPTIONS- NONE                                 **          00000480
         49   ***     DATES OF CODING AND REVISIONS-                                **          00000490
         50   ***        VERSION   DATE      REASON FOR VERSION ASSIGNMENT          **          00000500
         51   **           0     3/27/78        INITIAL DESIGN                      **          00000510
         52   **                                                                    **          00000520
         53   *********************************************************************          00000530
         54   ***  END PROLOGUE                                                     **          00000540
         55   **                                                                    **          00000550
         56   **MODULE MARES                                                        **          00000560
         57   **CALL RAEMSG TO WRITE ERROR MESSAGE                                  **          00000570
         58   **IF THE ERROR MESSAGE WAS NOT IN THE ERROR FILE                      **       1  00000580
         59   **      SET UP THE RAEMSG CALLING SEQUENCE TO WRITE                   **          00000590
         60   **           THE NOT FOUND MESSAGE                                    **          00000600
         61   **      CALL RAEMSG TO WRITE THE MESSAGE                              **          00000610
         62   **ENDIF                                                               **       2  00000620
         63   **IF AN ERROR OCCURRED                                                **       3  00000630
         64   **      SET ABORT TO THE APPROPRIATE VALUE FOR A READ                 **          00000640
         65   **           OR WRITE ERROR OR MESSAGE NOT FOUND                      **          00000650
         66   **ENDIF                                                               **       4  00000660
         67   **ENDMODULE  MARES                                                    **          00000670
         68   **                                                                    **          00000680
         69   ***  END PDL                                                          **          00000690
```

Figure 4-2. Prologue/PDL Report



```
                      AUTOMATED CODE AUDITOR
                     TEST PROCEDURE ANALYSIS

*****************************************************************
* DD *--INIT -*-                PARAMETER --          TEST CASES -
*PATH* CONSTR *                                   *1*2*3*4*5*6*7*8*
*****************************************************************
*   1 *IF-TRUE * THE ERROR MESSAGE WAS NOT IN TH   * * * * * * * * *
*   2 *IF-FALSE* THE ERROR MESSAGE WAS NOT IN TH   * * * * * * * * *
*   3 *IF-TRUE * AN ERROR OCCURRED                 * * * * * * * * *
*   4 *IF-FALSE* AN ERROR OCCURRED                 * * * * * * * * *
*****************************************************************
         MODULE= MARES             DATE=  10/26/78          PAGE    1
```

Figure 4-3. Test Specifications Chart

]

SECTION 5 - DIAGNOSTICS

Diagnostic messages generated by the Code Auditor are described in this section. Explanations of the probable cause and solutions are included in appropriate cases. The majority of these messages are related to analysis of Module Prologue and PDL.

ERROR NUMBER	MESSAGE TEXT	EXPLANATION
MA001 MA002	Invalid Control Card Keyword Invalid Module Name-Skipping to next dollar sign.	See Section 3.1 on control card format. A module name has been specified on a control card which is either less than two characters, more than eight characters, or contains an illegal character.
MA004	The following source module is empty, xxxxxx thus cannot be analyzed.	Input module xxxxxx contains no data.
MA005	Missing or invalid keyword on control card - skipping to next dollar sign.	See Section 3.1 on control card format. format.
MA007	Missing start control card: skipping to next dollar sign.	See Section 3.1 on control card format.
MA008	Missing TERM control card.	See Section 3.1 on control card format.
MA009	The following source module is not in the source PDS: xxxxxx	MODULE XXXXXX, specified to be analyzed, is not in the prologue/PDL library.
MA010	END PDL keyword not present: the entire PDL is presumed missing.	An example of PDL including the END PDL card appears in Figure 1-2.
MA011	IF keyword found before predecessor IF condition satisfied.	An IF keyword appears in PDL without a matching ENDIF.
MA012	DO-WHILE keyword found before predecessor DO-WHILE condition satisfied.	A DO-WHILE keyword has appeared in PDL without a matching ENDDO.
MA013	ENDDO found without a related DO-WHILE condition.	Self-explanatory
MA014	Number of diagnostic messages exceeds maximum number.	More than 100 diagnostics have been generated. No more will be generated for the module currently being analyzed.
MA015	Error code xxxxxx not in error message file	An attempt has been made to print diagnostic error message xxxxxx, which is not in the diagnostic error message library.

ERROR NUMBER	MESSAGE TEXT	EXPLANATION
MA016	Executive control table full-remainder of commands ignored.	More than 100 MODULE cards are in the control card file. The first 100 are recognized.
MA017	Control Cards were found beyond the TERM $ card and will be ignored.	See Section 3.1 on control card format.
MA018	Source text contains too many statements	More than 1000 statements have been input for a module prologue/PDL. This module will not be analyzed.
MA020	Predecessor DO-CASE end not satisfied.	A DO-CASE keyword has appeared in PDL without a matching ENDCASE.
MA021	ORIF is not a valid PDL keyword	ORIF is a JOVIAL keyword, but is not valid in PDL.
MA022	PD Table overflow	The main table used in creating test specification charts has overflowed.
MA024	Input cards exceed maximum of 106 - excess cards will be ignored.	Self-explanatory
MA025	Card contains an asterisk in CC 3 with no accompanying prologue keywords.	All cards containing keywords have an asterisk in column 3.
MA026	END prologue keyword not present. The entire prologue is presumed missing	Figure 1-1 contains a sample Module prologue.
MA027	END PDL keyword not present.	The entire PDL is presumed missing. Figure 1-2 contains sample PDL including the END PDL card.
MA028	Prologue analysis aborted: END PDL appears before end prologue keyword.	Input Modules should consist of a prologue followed by PDL.
MA029	Prologue and PDL cards must have comment delimeters in columns 1,2,65 and 66.	Valid comment delimeters are: asterisks, the character C, or single quotes.
MA-031	Text missing - skipping to next major prologue keyword.	Prologue text should have appeared where indicated by this message. Figure 1-1 contains a sample module prologue.

ERROR		
NUMBER	MESSAGE TEXT	EXPLANATION
MA032	One or both of these keywords is/are missing: usage class or usage restrictions.	See Figure 1-1 for a sample module prologue.
MA033	Duplicate keyword (or misplaced line of asterisks)-skipping to next major prologue keyword.	Major prologue keywords are those beginning in column eight. Also, the line of asterisks which begins the identification section is considered a major keyword.
MA034	Missing Minor prologue keyword associated with previous major keyword or possibly a missing asterisk in CC 3-skipping to next major keyword.	Minor prologue keywords begin in columns other than eight. All module prologue cards containing keywords must have an asterisk in column three.
MA035	Not a major prologue keyword (or missing * in cc3)-skipping to next major keyword.	A major prologue keyword should have appeared at the location indicated. Major keywords begin in column 8. Additionally, the line of asterisks beginning the identification section is considered to be a major keyword.
MA036	The line of asterisks which begins the identification section was never found.	Self-explanatory
MA037	Limit of Prologue Matrix exceeded: Remainder of prologue will be ignored.	The prologue being analyzed contains **duplicate** keywords and text beyond reasonable limits. The portion of the prologue preceeding this message will be analyzed.
MA038	There should be a line of asterisks immediately preceeding 'end prologue'.	'END Prologue" refers to the END Prologue card.
MA039	Missing Minor keyword(s) in module identification section.	Minor keywords begin in columns other than eight. The only major keyword in the module identification section is the initial line of asterisks. Strange, but true.
MA040	Major Prologue keyword not locatable. Called By-	Self-explanatory
MA041	Major Prologue Keyword not locatable. Calls-	Self-explanatory
MA042	Major Prologue keyword not locatable. Core Requirements	Self-explanatory

ERROR NUMBER	MESSAGE TEXT	EXPLANATION
MA043	Major Prologue Keyword not locatable: Dates of Coding & Revisions-	Self-explanatory
MA044	Major Prologue Keyword not locatable: English name-	Self-explanatory
MA045	Major Prologue Keyword not locatable: Error handling-	Self-explanatory
MA046	Major Prologue Keyword not locatable: Input data	Self-explanatory
MA047	Major Prologue Keyword not locatable: Intermediate Data-	Self-explanatory
MA048	Major Prologue Keyword not locatable: Limitations/ Assumptions-	Self-explanatory
MA049	Major Prologue Keyword not locatable: Location.	Self-explanatory
MA050	Major Prologue Keyword not locatable - Method narrative-	Self-explanatory
MA051	Major Prologue Keyword not locatable: Module Type	Self-explanatory
MA052	Major Prologue Keyword not locatable: Output data-	Self-explanatory
MA053	Major Prologue Keyword not locatable: Purpose-	Self-explanatory
MA054	Major Prologue Keyword not locatable: Special Techniques	Self-explanatory
MA055	Major Prologue Keyword not locatable: Timing estimate-	Self-explanatory